LISTING OF CLAIMS:

1. (Currently Amended) A method comprising:

utilizing first and second virtual machine queues associated with respective first and second virtual machines to communicate between the virtual machines; and

updating a page table in a processor by placing a page associated with the first virtual machine in an address space associated with the second virtual machine.

2. (Canceled)

3. (Currently Amended) The method according to claim ~~2~~1, wherein updating includes:

placing at least one of data and an address associated with the page into a first virtual machine control structure associated with the first virtual machine;

exiting the first virtual machine;

placing the at least one of data and address into the second virtual machine queue; and

dequeueing the second virtual machine queue.

4. (Original) The method according to claim 3, wherein dequeueing includes:

reading the at least one of data and address into a second virtual machine control structure associated with the second virtual machine; and

storing the at least one of data and address into the address space associated with the second virtual machine.

5. (Original) The method according to claim 3, wherein the page contains a message and the method further comprises:

processing the message within the second virtual machine.

6. (Original) The method according to claim 3, wherein exiting occurs immediately after placing the at least one of data and an address associated with the page into the first virtual machine control structure.

7. (Original) The method according to claim 1, further comprising:

conveying identification information associated with the first and second virtual machines between the first and second virtual machines via the first and second virtual machine queues.

8. (Currently Amended) A computer system comprising:

at least one processor; and

a computer readable memory comprising program instructions, executable by the at least one processor, for:

first and second virtual machines;

a first virtual machine control structure associated with the first virtual machine, the first virtual machine control structure having a first virtual machine queue adapted to enqueue and dequeue a message;

a second virtual machine control structure associated with the second virtual machine, the second virtual machine control structure having a second virtual machine queue adapted to enqueue and dequeue a message; and

a virtual machine monitor coupled to the first and second virtual machines and to the first and second virtual machine control structures, the virtual machine monitor adapted to supervise communication between the first and second virtual machines.

9. (Original) The computer system according to claim 8, wherein the virtual machine monitor is further adapted to update a page table in a processor by placing a page associated with the first virtual machine in an address space associated with the second virtual machine.

10. (Original) The computer system according to claim 9, wherein the virtual machine monitor is further adapted to place at least one of data and an address associated with the page into the first virtual machine control structure.

11. (Original) The computer system according to claim 10, wherein the virtual machine monitor is further adapted to cause a virtual machine exit.

12. (Original) The computer system according to claim 11, wherein the virtual machine monitor is further adapted to place at least one of data and an address into the second virtual machine queue.

13. (Original) The computer system according to claim 12, wherein the second virtual machine is adapted to process the page.

14. (Original) The computer system according to claim 8, wherein the virtual machine monitor is further adapted to convey identification information associated with the first and second virtual machines between the first and second virtual machines via the first and second virtual machine queues.

15. (Currently Amended) A computer readable memory containing program instructions that, when executed by a processor, cause the processor to:

utilize first and second virtual machine queues associated with respective first and second virtual machines to communicate between the virtual machines; and

update a page table in a processor by placing a page associated with the first virtual machine in an address space associated with the second virtual machine.

16. (Canceled)

17. (Original) The computer readable memory according to claim 16, containing further program instructions that, when executed by a processor, cause the processor to:

place at least one of data and an address associated with the page into a first virtual machine control structure associated with the first virtual machine;

exit the first virtual machine;

place the at least one of data and address into the second virtual machine queue; and

dequeue the second virtual machine queue.

18. (Original) The computer readable memory according to claim 16, containing further program instructions that, when executed by a processor, cause the processor to:

read the at least one of data and address into a second virtual machine control structure associated with the second virtual machine; and

store the at least one of data and address into the address space associated with the second virtual machine.

19. (Original) The computer readable memory according to claim 17, wherein the page contains a message, and wherein the computer readable memory contains further program instructions that, when executed by a processor, cause the processor to:

process the message within the second virtual machine.

20. (Original) The computer readable memory according to claim 15, containing further program instructions that, when executed by a processor, cause the processor to:

convey identification information associated with the first and second virtual machines between the first and second virtual machines via the first and second virtual machine queues.